



Cours d'intégration

3W Academy

HTML 4. Les formulaires

4. Les formulaires

4.1. Présentation

4.2. la balise form

4.3. Les champs

4.3.1. la balise input

4.3.1.1. Pour mettre une valeur initiale

4.3.1.2. Pour rendre le champ est obligatoire

4.3.1.3. Les autres attributs

4.3.1.3. Les différents types

4.3.1.4. Exemple de groupe de boutons radio

4.3.1.4. Exemple de groupe de cases à cocher

4.3.2. la balise textarea

4.3.3. la balise select

4.3.3.1 Un exemple simple

4.3.3.1 Un choix multiple

4.3.3.1 Une organisation par groupes

4.4. Les libellés

4.4. Les groupes de champs

4.5. Les boutons

4. Les formulaires

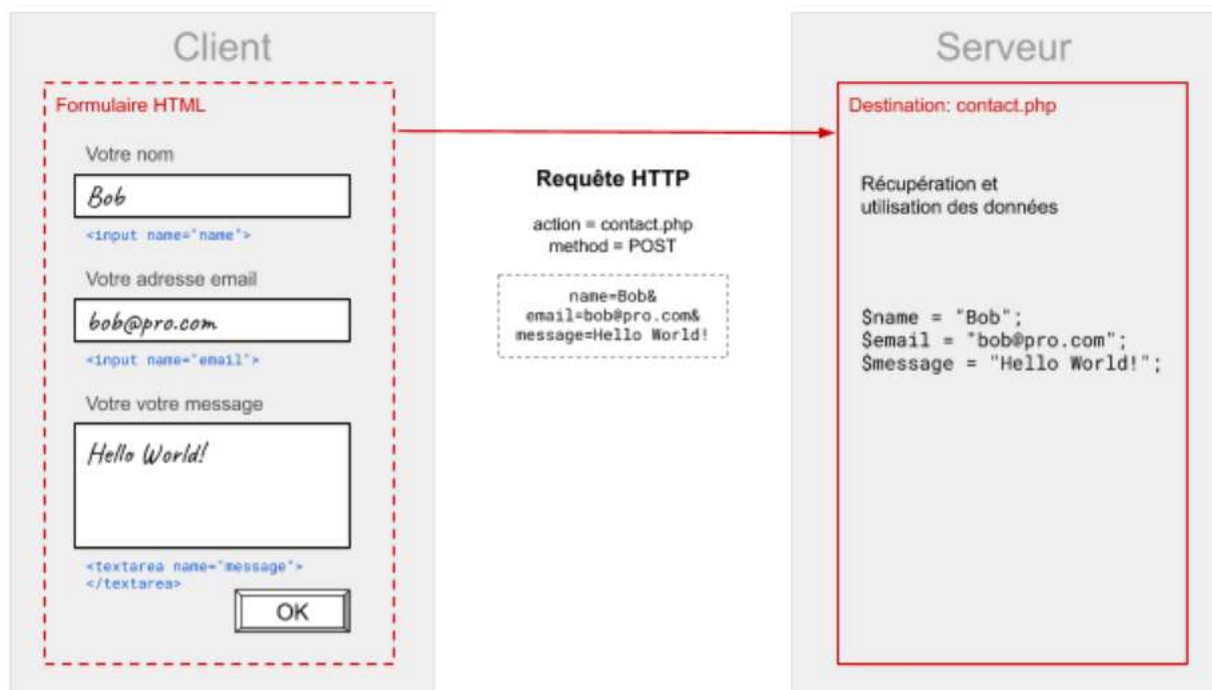
4.1. Présentation

Un formulaire HTML permet d'envoyer des données au serveur sous la forme de variables (paire nom / valeur) appelés champs. Un formulaire est un ensemble de champs, il peut donc y avoir plusieurs formulaires dans une même page Web. Chaque formulaire est indépendant et possède l'url du fichier de destination des données dans un attribut *action*.

Il existe plusieurs méthodes d'envoi des données (GET et POST).

- Les données envoyées par la méthode GET sont visibles dans l'url dans la barre d'adresse du navigateur.
- Les données envoyées par la méthode POST, sont encapsulées dans l'entête HTTP de la requête (mais restent accessibles via l'inspecteur dans la partie réseau).

L'envoi des données via un formulaire peut être sécurisé en utilisant le protocole HTTPS, à configurer sur le serveur.



4.2. la balise form

Elle doit contenir l'ensemble des champs du formulaire et à minima, l'url du fichier de destination des données dans l'attribut action ainsi que la méthode utilisée.

```
<form action="contact.php" method="post">  
    ...  
</form>
```

Dans le cas où l'on souhaite associer des fichiers, on peut préciser l'encodage de ces derniers.

```
<form action="contact.php" method="post" enctype="multipart/form-data">  
    ...  
</form>
```

4.3. Les champs

4.3.1. la balise input

C'est une balise auto-fermante. Elle doit contenir, à minima, le type ainsi que le l'attribut name qui correspond au nom de la variable côté serveur.

```
<input type="text" name="city">
```

4.3.1.1. Pour mettre une valeur initiale

```
<input type="text" name="city" value="Paris">
```

4.3.1.2. Pour rendre le champ obligatoire

```
<input type="text" name="city" required>
```

4.3.1.3. Les autres attributs

readonly	le champ est en lecture seule (sélection possible, édition impossible)
disabled	le champ est désactivé (sélection et édition impossibles)
placeholder	affiche un libellé à la place de la valeur si celle-ci est nulle
minlength	définit le nombre de caractères minimum
maxlength	définit le nombre de caractères maximum
tabindex	définit l'ordre des champs lors de la navigation via la touche TAB (focus)
autofocus	place le focus automatiquement sur ce champ lors du chargement
min	utilisé avec les types number et range, précise la valeur minimum
max	utilisé avec les types number et range, précise la valeur maximum
step	utilisé avec les types number et range, précise l'incrément de progression entre les valeurs min et max

4.3.1.3. Les différents types

text	champ texte simple (par défaut) <input type="text" value="Text Input"/>
password	champ dont la valeur est un mot de passe, elle est cachée, remplacée par des points pour éviter d'être lue par d'autres au moment de la saisie <input type="password" value="...."/>
radio	choix unique parmi plusieurs (OU) <input checked="" type="radio"/> Option 1 <input type="radio"/> Option 2 <input type="radio"/> Option 3
checkbox	choix multiple parmi plusieurs (ET) <input checked="" type="checkbox"/> Choice A <input type="checkbox"/> Choice B <input type="checkbox"/> Choice C

file	champ permettant de sélectionner un ou plusieurs fichiers sur son ordinateur <div>Choisir un fichier</div> Aucun fichier choisi																																										
hidden	champ caché, très utilisé pour transmettre au serveur des informations liées à la sécurité ou à la session en cours																																										
search	champ dédié à la recherche <div>term</div>																																										
number	champ permettant uniquement la saisie d'un nombre <div>Enter a Number</div>																																										
range	champ permettant de choisir une valeur dans une plage de valeurs (avec une réglette coulissante) <div></div>																																										
color	champ permettant uniquement la saisie d'une couleur <div></div>																																										
date	champ permettant de choisir une date <div>01 / 01 / 1970</div> <div>janvier 1970</div> <table><tr><td>lun.</td><td>mar.</td><td>mer.</td><td>jeu.</td><td>ven.</td><td>sam.</td><td>dim.</td></tr><tr><td>29</td><td>30</td><td>31</td><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td></tr><tr><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td></tr><tr><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td></tr><tr><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td><td>1</td></tr></table>	lun.	mar.	mer.	jeu.	ven.	sam.	dim.	29	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1
lun.	mar.	mer.	jeu.	ven.	sam.	dim.																																					
29	30	31	1	2	3	4																																					
5	6	7	8	9	10	11																																					
12	13	14	15	16	17	18																																					
19	20	21	22	23	24	25																																					
26	27	28	29	30	31	1																																					
email	champ permettant uniquement la saisie d'une adresse email <div>name@email.com</div>																																										
url	champ permettant uniquement la saisie d'une URL <div>http://127.0.0.1</div>																																										
tel	champ dédié à la saisie d'un numéro de téléphone <div>(999) 999-9999</div>																																										

4.3.1.4. Exemple de groupe de boutons radio

Pour que les boutons fonctionnent ensemble (si on coche un des boutons, cela doit décocher les autres), ils doivent tous avoir le même *name*. Ici la valeur bleu est pré-sélectionné.

```
<label><input type="radio" name="color" value="r">Rouge</label>
<label><input type="radio" name="color" value="g">Vert</label>
<label><input type="radio" name="color" value="b" checked>Bleu</label>
```

4.3.1.4. Exemple de groupe de cases à cocher

Ici, on peut mettre un *name* avec des crochets pour que le PHP récupère les valeurs sélectionnées côté serveur sous la forme d'un tableau. Mettre toujours 1 comme valeur (1 = oui = true). Ici les valeurs vert et bleu sont pré-sélectionnés.

```
<label><input type="checkbox" name="colors[r]" value="1">Rouge</label>
<label><input type="checkbox" name="colors[g]" value="1"
checked>Vert</label>
<label><input type="checkbox" name="colors[b]" value="1"
checked>Bleu</label>
```

4.3.2. la balise textarea

Si l'on souhaite un champ de saisie multilignes, il faut utiliser la balise `textarea`.

A screenshot of a web browser showing a text area with the placeholder text "Enter your message here". The text area is rectangular with a thin border and a small cursor icon in the bottom right corner.

Par exemple:

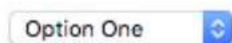
```
<textarea name="name" placeholder="Veuillez saisir votre
adresse"></textarea>
```



Attention le contenu initial doit se placer entre la balise ouvrante et la balise fermante. Si le contenu est vide, les balises ouvrante et fermante doivent être collées (sinon, il se peut que vous envoyez des espaces comme valeur au serveur).

4.3.3. la balise select

Si l'on souhaite afficher un choix unique (comme les boutons radio) ou multiple (comme les cases à cocher) sous la forme d'un menu, on peut utiliser la balise *select*.



4.3.3.1 Un exemple simple

Ici les carottes sont présélectionnées.

```
<select name="panier">
  <option value="fraises">Fraises</option>
  <option value="bananes">Bananes</option>
  <option value="carottes" selected>Carottes</option>
  <option value="navets">Navets</option>
</select>
```

4.3.3.1 Un choix multiple

Ici les bananes et les carottes sont présélectionnées. L'attribut `size` permet d'indiquer le nombre de lignes affichées simultanément.

```
<select name="panier" multiple size="3">
  <option value="fraises">Fraises</option>
  <option value="bananes" selected>Bananes</option>
  <option value="carottes" selected>Carottes</option>
  <option value="navets">Navets</option>
</select>
```

4.3.3.1 Une organisation par groupes

Pour organiser les options par groupes, on peut les regrouper dans des balises *optgroup*.

```
<select name="panier" multiple size="3">
  <optgroup label="Fruits">
    <option value="fraises">Fraises</option>
    <option value="bananes" selected>Bananes</option>
  </optgroup>
  <optgroup label="Légumes">
    <option value="carottes" selected>Carottes</option>
    <option value="navets">Navets</option>
  </optgroup>
</select>
```

4.4. Les libellés

Pour afficher correctement un label d'un champ, on utilise la balise *label*. L'association se fait via l'attribut *for* du label qui doit correspondre à la valeur de l'*ID* du champ.

```
<label for="city-id">Votre ville</label>
<input type="text" name="city" id="city-id">
```

Dans l'exemple, on a un *ID* qui est différent du *name* mais ils peuvent être identique.

L'association permet par exemple de cocher une case à cocher directement en cliquant sur le libellé, ce qui améliore l'accessibilité et l'ergonomie du formulaire.

Dans le cas des cases à cocher ou des boutons radio, il est possible de placer directement la balise *input* dans la balise et d'omettre le couple *for* / *ID*:

```
<label>
  <input type="checkbox" name="conditions">
  J'ai pris connaissance des conditions d'utilisation.
</label>
```

4.4. Les groupes de champs

Pour organiser les champs par groupes, on peut utiliser le couple de balises *fieldset* / *legend*. La balise *legend* n'est pas obligatoire mais si elle est présente, elle doit être le premier élément du *fieldset*.

```
<fieldset>
  <legend>Titre du groupe de champs</legend>
  ...
</fieldset>
```

4.5. Les boutons

Pour envoyer les données au serveur, il faut valider le formulaire via un bouton. On peut utiliser la balise `input` avec le type "submit" mais il est préférable d'utiliser la balise `button`. Elle est composée d'une balise ouvrante et d'une balise fermante. Le libellé se place entre les deux et peut contenir d'autres éléments comme une image.

Elle possède également un attribut `type` qui peut prendre 3 valeurs:

submit	les données sont envoyées au serveur à l'adresse spécifiée dans l'attribut <i>action</i> de la balise <i>form</i>
button	aucune donnée n'est envoyée, très souvent utilisé conjointement avec du JavaScript pour effectuer des actions personnalisées sur le formulaire
reset	très peu utilisé, réinitialise tous les champs du formulaire en leur donnant leur valeur par défaut

Un exemple de bouton d'envoi des données avec l'attribut `type="submit"`:

```
<button type="submit">Envoyer les informations</button>
```